NASA/CR-2002-211451
ICASE Report No. 2002-3

Loops of Superexponential Lengths in One-rule String Rewriting

Alfons Geser
ICASE, Hampton, Virginia

February 2002

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATIONS. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized data bases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***

- Email your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA STI Help Desk at (301) 621-0134

- Telephone the NASA STI Help Desk at (301) 621-0390

- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076-1320

NASA/CR-2002-211451
ICASE Report No. 2002-3

Loops of Superexponential Lengths in One-rule String Rewriting

Alfons Geser
ICASE, Hampton, Virginia

ICASE
NASA Langley Research Center
Hampton, Virginia

Operated by Universities Space Research Association

February 2002

Available from the following:

NASA Center for AeroSpace Information (CASI)
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 487-4650

LOOPS OF SUPEREXPONENTIAL LENGTHS IN ONE-RULE STRING REWRITING*

ALFONS GESER†

Abstract. Loops are the most frequent cause of non-termination in string rewriting. In the general case, non-terminating, non-looping string rewriting systems exist, and the uniform termination problem is undecidable. For rewriting with only one string rewriting rule, it is unknown whether non-terminating, non-looping systems exist and whether uniform termination is decidable. If in the one-rule case, non-termination is equivalent to the existence of loops, as McNaughton conjectures, then a decision procedure for the existence of loops also solves the uniform termination problem. As the existence of loops of bounded lengths is decidable, the question is raised how long shortest loops may be. We show that string rewriting rules exist whose shortest loops have superexponential lengths in the size of the rule.

Key words. string rewriting, semi-Thue system, uniform termination, termination, loop, one-rule, single-rule

Subject classification. Computer Science

1. Introduction. Uniform termination, i.e. the non-existence of an infinite reduction sequence, is an undecidable property of string rewriting systems (SRSs) [8], even if they comprise only three rules [13]. It is open whether uniform termination is decidable for SRSs with less than three rules.

An SRS admits a *loop* if there is a reduction of the form $u \to^+ sut$. Every looping SRS is non-terminating. The converse does not hold, even for two-rule SRSs [6]. McNaughton [15] conjectures that every one-rule non-terminating SRS admits a loop.

If McNaughton's conjecture holds, and if the existence of loops is decidable for one-rule SRSs, then the uniform termination of one-rule SRSs is decidable. Existence of loops of *bounded length* is decidable [6]. This immediately raises the question whether there is an algorithm that outputs upper bounds of lengths of shortest loops. On this account it is most interesting how long shortest loops can be.

The purpose of this note is to prove that there are one-rule SRSs that admit loops of superexponential lengths in the size of the rule, but no shorter loops. This is in harsh contrast to the common belief that loops are simple. Specifically, we prove the following result.

THEOREM 1.1. *For all $p \geq 2q$, $q \geq 1$, $r \geq 2$, the string rewriting rule*

$$R = \{10^p \to 0^p 1^r 0^q\}$$

admits loops of length $1 + \sum_{i=0}^{\ell-1} r^i$ where $\ell = \lceil \frac{p}{q} \rceil$ but no shorter loops.

Theorem 1.1 follows immediately from Lemmas 4.6 and 6.17, which we will prove below.

By choosing $q = 1$ and keeping $p \geq 2$ fixed, we get a family of rules where the shortest length of loops is polynomial in r with degree $p - 1$. By choosing $q = 1$ and keeping $r \geq 2$ fixed, we get shortest loop lengths exponential in p with base r. By choosing $q = 1$ and $r = p$ the minimal loop length is greater than p^{p-1}. This shows the claimed superexponential growth.

*This work was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-97046 while the author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-2199, USA.

†Address: ICASE, Mail Stop 132C, NASA Langley Research Center, Hampton, VA 23681. Email: geser@icase.edu

The paper is organized as follows. We will show: in Section 3 that each R has a loop; in Section 4 that the length of the loop is as claimed; in Section 5 that the start string of a shortest loop has a special shape; and in Section 6 that these strings initiate no shorter loops.

2. Preliminaries. We assume that the reader is familiar with termination of string rewriting. SRSs are also called *semi-Thue systems.*

For an introduction to string rewriting see Book and Otto [1] or Jantzen [9]. The study of termination in one-rule string rewriting has been initiated by Kurth in his thesis [11]. Further work includes McNaughton [14, 15, 16], Senizergues [19], Kobayashi et al. [20, 10], and Zantema and Geser [6, 21, 4, 5]. Since SRSs can be encoded as term rewriting systems where letters are unary function symbols, the results of termination of term rewriting [3] apply.

An SRS R is a set of *string rewriting rules*, i.e. pairs of strings denoted as $u \to v$. The reduction step relation, also denoted by $\to$, is defined by $sut \to svt$ for all strings s, t and string rewriting rules $u \to v$. Here st denotes the *concatenation* of strings s and t.

A *loop* is a reduction of the form $t \to^+ utv$ where u, v are strings. An SRS R is said to *admit a loop* if a loop $t \to^+ utv$ exists.

The string t is also called a *prefix*, u a *suffix* of tu. Any string utv is said to contain t as a *factor*. The set of *overlaps* of a string u with a string v is defined by

$$\mathrm{OVL}(u, v) = \{w \in \Sigma^+ \mid u = u'w, v = wv', u'v' \neq \varepsilon, u', v' \in \Sigma^*\} .$$

3. The Rule Admits a Loop. Throughout this paper we assume strings over the two-letter alphabet $\{0, 1\}$, and we speak about one-rule SRSs $R = \{10^p \to 0^p1^r0^q\}$, for some $p \geq 2q$, $q \geq 1$, $r \geq 2$. In this section we show that each R has a loop.

DEFINITION 3.1. *Let strings t_i, $i \geq 0$ be defined recursively by*

$$t_0 = 1,$$
$$t_{i+1} = 0^q t_i^r .$$

The following lemma is crucial for the proof that R admits loops.

LEMMA 3.2. $t_k^m 0^p \to^* 0^{p-q} t_{k+1}^m 0^q$ *holds for all* $k, m \geq 0$.

Proof. Proof by induction on (k, m) ordered lexicographically. The case $m = 0$ is trivial, so assume $m > 0$. Case 1: $k = 0$. Then

$$t_0^m 0^p = t_0^{m-1} 10^p \to t_0^{m-1} 0^p 1^r 0^q \to^* 0^{p-q} t_1^{m-1} 0^q 1^r 0^q = 0^{p-q} t_1^m 0^q,$$

by definition of t_0, inductive hypothesis for $(k, m-1)$, and definition of t_1, respectively. Case 2: $k > 0$. Then

$$t_k^m 0^p = t_k^{m-1} 0^q t_{k-1}^r 0^p \to^* t_k^{m-1} 0^q 0^{p-q} t_k^r 0^q \to^* 0^{p-q} t_{k+1}^{m-1} 0^q t_k^r 0^q = 0^{p-q} t_{k+1}^m 0^q,$$

by definition of t_k, inductive hypothesis for $(k-1, r)$, inductive hypothesis for $(k, m-1)$, and definition of t_{k+1}, respectively. □

By definition t_k is a factor of t_{k+1}. Now if $t_k 0^p$ is a factor of $t_{k+1} 0^q$ then we have a loop. To this end k has to be great enough.

EXAMPLE 1. *Let $p = 2$, $q = 1$, $r = 2$. Then $R = \{100 \to 00110\}$. We have $t_0 = 1$, $t_1 = 0t_0t_0 = 011$, $t_2 = 0t_1t_1 = 0011011$, $t_3 = 0t_2t_2 = 000110110011011$, and so forth. The string $t_0 0^p = 100$ is not a factor* *of* $t_1 0^q = 0110$. *Neither is* $t_1 0^p = 01100$ *a factor of* $t_2 0^q = 00110110$. *However* $t_3 0^q = 0\underline{00110110}0110110$ *contains* $t_2 0^p$ *as a factor at the underlined occurrence.*

The problem is traced back to finding a factor 0^p within t_k. The following property of t_i is the key to the solution.

LEMMA 3.3. *For all* $k \geq 0$, *the following hold:*

1. 0^{qk} *is a prefix of* t_k.
2. 0^{qk+1} *is not a factor of* t_k.

Proof. Straightforward induction on k. □

If k is chosen great enough then 0^p fits into 0^{qk}. Let $\lceil x \rceil$ denote the least integer i such that $i \geq x$.

LEMMA 3.4. *Let* $\ell = \lceil \frac{p}{q} \rceil$. *Then* $t_\ell 0^p \to^* 0^{p-q} t_{\ell+1} 0^q$ *is a loop.*

Proof. By Lemma 3.2 for $m = 1$ we get a reduction

$$t_\ell 0^p \to^* 0^{p-q} t_{\ell+1} 0^q \ . \tag{3.1}$$

Now suppose that $\ell = \lceil \frac{p}{q} \rceil$, whence $q\ell \geq p$. The following analysis shows that in this case Reduction (3.1) indeed forms a loop, i.e. that its left hand side $t_\ell 0^p$ is a factor of its right hand side, $0^{p-q} t_{\ell+1} 0^q$. For some string w we get

$$0^{p-q} t_{\ell+1} 0^q \ = \ 0^{p-q} 0^q t_\ell^r 0^q \ = \ 0^{p-q} 0^q t_\ell^{r-2} t_\ell t_\ell 0^q \ = \ 0^{p-q} 0^q t_\ell^{r-2} t_\ell 0^{q\ell} w 0^q \ = \ 0^{p-q} 0^q t_\ell^{r-2} \underline{t_\ell 0^p} 0^{q\ell - p} w 0^q$$

by definition of $t_{\ell+1}$, the premise $r \geq 2$, Lemma 3.3, and the property $q\ell \geq p$, respectively. The occurrence of $t_\ell 0^p$ is underlined. □

4. The Loop has Superexponential Length. Now let us calculate the lengths of reductions $t_\ell 0^p \to^* 0^{p-q} t_{\ell+1} 0^q$ of Lemma 3.4. We start with a recursive specification of the lengths of reductions of Lemma 3.2. Let $\mathbb{N}$ denote the set of non-negative integers.

DEFINITION 4.1. *The function* $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ *is defined recursively by*

$$f(k, 0) = 0,$$
$$f(0, m+1) = 1 + f(0, m),$$
$$f(k+1, m+1) = f(k, r) + f(k+1, m) \ .$$

The following properties of f are obvious.

PROPOSITION 4.2. f *is well-defined and a total function.*

PROPOSITION 4.3. $f(k, m)$ *equals the length of the reduction* $t_k^m 0^p \to^* 0^{p-q} t_{k+1}^m 0^q$ *constructed in Lemma 3.2.*

It is straightforward to check that the following non-recursive definition of f satisfies Definition 4.1:

PROPOSITION 4.4. $f(k, m) = mr^k$ *for all* $k, m \geq 0$.

From Propositions 4.3 and 4.4 we get immediately:

PROPOSITION 4.5. *The length of the reduction in Lemma 3.4 is* r^ℓ.

The length of the loop in Lemma 3.4 is not yet minimal. A refinement leads to the following shorter loop. We will prove its minimality in the subsequent sections.

LEMMA 4.6. *Let* $\ell = \lceil \frac{p}{q} \rceil$. *Then there is a loop*

$$1 0^{(p-q)(\ell+1)+q} \to^n 0^p 1^{r-1} 0^{(p-q)\ell} t_\ell 0^q$$

of length $n = 1 + \sum_{i=0}^{\ell-1} r^i$.

Proof. By Lemma 3.2 and Proposition 4.3 we have the reduction

$$\begin{aligned}
10^{(p-q)(\ell+1)+q} &\to \\
0^p1^{r-1}t_0 0^{(p-q)\ell+q} &\to^{r^0} \\
0^p1^{r-1}0^{p-q}t_1 0^{(p-q)(\ell-1)+q} &\to^{r^1} \\
&\vdots \\
0^p1^{r-1}0^{(p-q)(\ell-1)}t_{\ell-1}0^{(p-q)+q} &\to^{r^{\ell-1}} 0^p1^{r-2}\underline{10^{(p-q)\ell}t_\ell} 0^q \ .
\end{aligned}$$

Now t_ℓ has a prefix $0^{q\ell}$ by Lemma 3.3, and so a prefix 0^p by definition of ℓ. Together with the underlined string this forms a reoccurrence of the initial string, $10^{(p-q)(\ell+1)+q} = 10^{(p-q)\ell}0^p$, as a factor in the final string. So the given reduction is indeed a loop. □

5. How Shortest Loops Start. To prove that there are no loops shorter than those of Lemma 4.6, we first restrict the set of strings that may initiate shortest loops. To this end we employ the fact that the existence of loops is characterized by the existence of looping forward closures. Forward closures [12, 2] are restricted reductions. The following characterization of forward closures by Hermann is convenient.

DEFINITION 5.1 (Forward Closure [12, 2, 7]). *The set of* forward closures *of an SRS R is the least set* $\mathrm{FC}(R)$ *of R-reductions such that*

fc1. if $(l \to r) \in R$ then $(l \to r) \in \mathrm{FC}(R)$,

fc2. if $(s_1 \to^+ t_1'x) \in \mathrm{FC}(R)$ and $(xl_2' \to r_2) \in R$ such that $x \neq \varepsilon$ then $(s_1l_2' \to^+ t_1'xl_2' \to^+ t_1'r_2) \in \mathrm{FC}(R)$,

fc3. if $(s_1 \to^+ t_1'l_2t_1'') \in \mathrm{FC}(R)$ and $(l_2 \to r_2) \in R$ then $(s_1 \to^+ t_1'l_2t_1'' \to^+ t_1'r_2t_1'') \in \mathrm{FC}(R)$.

We call a forward closure of the form $s \to^+ usv$ a *looping* forward closure.

THEOREM 5.2 ([6]). *An SRS admits a loop if and only if it has a looping forward closure. Moreover if there is a loop of length n then there is a looping forward closure of length at most n.*

LEMMA 5.3. *Every forward closure of R has the form $10^{(p-q)k+q} \to^* w1^r0^q$ for some $k \geq 1$ and some string w.*

Proof. By induction on the definition of forward closure. Case *fc1* is trivial. For Case *fc2*, let $s_1 = 10^{(p-q)k+q}$, $t_1'x = w1^r0^q$, $xl_2' = 10^p$, $r_2 = 0^p1^r0^q$. Observe that x must be $x = 10^q$. This implies $t_1' = w1^{r-1}$, $l_2' = 0^{p-q}$ and we get

$$s_1l_2' = 10^{(p-q)k+q}0^{p-q} = 10^{(p-q)(k+1)+q} \to^* w1^{r-1}0^p1^r0^q = t_1'r_2$$

as the composed forward closure. It has the claimed form.

Case *fc3*: Let $s_1 = 10^{(p-q)k+q}$, $t_1'l_2t_1'' = w1^r0^q$, $l_2 = 10^p$, $r_2 = 0^p1^r0^q$. By $p > q$, l_2 cannot be a factor of 1^r0^q. Nor can it left overlap with it: $\mathrm{OVL}(l_2, 1^r0^q) = \emptyset$. Therefore t_1'' is longer than 1^r0^q. In other words a string w' exists such that $t_1'' = w'1^r0^q$. Hence $w = t_1'l_2w'$ and the composed forward closure is

$$s_1 = 10^{(p-q)k+q} \to^* t_1'r_2w'1^r0^q = t_1'r_2t_1''$$

which has the claimed form. □

Next we show that a forward closure can only issue an infinite reduction, and so a loop, if its left hand side is large enough.

LEMMA 5.4. $0^{(p-q)k+q}t_k0^q$ *is irreducible for all $k \leq \lceil \frac{p}{q} \rceil$.*

Proof. Suppose that $0^{(p-q)k+q}t_k0^q$ is reducible. Then t_k is reducible; but then t_{k-1} contains a factor 0^p; by Lemma 3.3 then $q(k-1) \geq p$; so $k \geq 1 + \lceil \frac{p}{q} \rceil$. □

THEOREM 5.5 ([18]). *Let R be non-overlapping and let s be an arbitrary string. Then s has an infinite reduction if and only if all reductions starting from s can be prolonged infinitely.*

LEMMA 5.6. *If $10^{(p-q)k+q}$ issues an infinite reduction then $k \geq 1 + \lceil \frac{p}{q} \rceil$.*

Proof. First we observe that R is non-overlapping, i.e. its left hand side, 10^p, has no overlap with itself: $\mathrm{OVL}(10^p, 10^p) = \emptyset$. Now there is a reduction $s = 10^{(p-q)k+q} \to^* 0^{(p-q)k} t_k 0^q = s'$ by Lemma 3.2 applied k times for $m = 1$. If $k \leq \lceil \frac{p}{q} \rceil$ then this reduction cannot be prolonged as its final string, s', is irreducible by Lemma 5.4. By Theorem 5.5 therefore s issues no infinite reduction for $k \leq \lceil \frac{p}{q} \rceil$. □

6. Shorter Loops Do Not Exist. We still have to prove that strings $10^{k(p-q)+q}$, $k \geq 1 + \ell = 1 + \lceil \frac{p}{q} \rceil$, initiate no loops shorter than $1 + \sum_{i=0}^{\ell-1} r^i$.

First let us switch from strings $s \in \{0,1\}^*$ to their tuple representation $T(s) \in \mathbb{N}^*$. Please note that our notion of tuple representation differs from the literature [17, 11].

DEFINITION 6.1. *A string $s \in \{0,1\}^*$ of the form*

$$s = 0^{x_0(p-q)+y_0 q} 1 0^{x_1(p-q)+y_1 q} \ldots 1 0^{x_k(p-q)+y_k q}$$

for some $k, x_0, \ldots, x_k \in \mathbb{N}$ and $0 \leq y_0, \ldots, y_k \leq \ell - 1$ is said to have a tuple representation

$$T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k) \ .$$

The guard $y_i \leq \ell - 1$, which is equivalent to $y_i q < p - q$, ensures that the x_i and y_i are uniquely given by $x_i(p-q) + y_i q$. Some strings over the alphabet $\{0,1\}$ may have no tuple representation, e.g. 0 has no tuple representation if $p = 4$, $q = 2$. For our purposes, however, it is reassuring to know that $10^{k(p-q)+q}$ has a tuple representation for any k and that certain rewrite steps preserve the existence of tuple representation.

We will conveniently speak about rewriting steps at position m:

DEFINITION 6.2. *Let s have a tuple representation, $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$, and let $0 \leq m \leq k-1$. Then $s \to_m s'$ if $x_{m+1} > 0$, $y_{m+1} > 0$, and*

$$s' = 0^{x_0(p-q)+y_0 q} 1 \ldots 1 0^{x_m(p-q)+y_m q} \underline{0^p 1^r 0^q} 0^{(x_{m+1}-1)(p-q)+(y_{m+1}-1)q} \ldots 1 0^{x_k(p-q)+y_k q} \ .$$

PROPOSITION 6.3. *If s has a tuple representation then $s \to s'$ if and only if $s \to_m s'$ for some $0 \leq m \leq k-1$.*

DEFINITION 6.4. *Let $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$ and let $0 \leq m \leq k-1$. Then a rewrite step $s \to_m s'$ is called* ordinary *if $y_m < \ell - 1$. Else the step is called* extraordinary. *A reduction is called ordinary if every step is ordinary. An extraordinary reduction has at least one extraordinary step.*

Ordinary rewrite steps preserve the existence of tuple representation:

PROPOSITION 6.5. *Let $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$, let $0 \leq m \leq k-1$, and let $s \to_m s'$ be ordinary. Then s' has the tuple representation*

$$T(s') = (x_0, \ldots, x_{m-1}, x_m + 1, \underbrace{0, \ldots, 0}_{r-1}, x_{m+1} - 1, x_{m+2}, \ldots, x_k;$$
$$y_0, \ldots, y_{m-1}, y_m + 1, \underbrace{0, \ldots, 0}_{r-1}, y_{m+1}, y_{m+2}, \ldots, y_k) \ .$$

In contrast, extraordinary steps may create strings that have no tuple representation.

EXAMPLE 2. *Let s have the tuple representation $T(s) = (2, 1; \ell - 2, 1)$, and let $m = 0$. Then we have*

$$s = 0^{2(p-q)+(\ell-2)q} 1 0^p \to_m 0^{3(p-q)+(\ell-1)q} 1 0^q = 0^{4(p-q)+\ell q - p} 1 0^q \ .$$

For $p = 5, q = 2$ we get $\ell = 3$ and $\ell q - p = 1$ which has no representation as an integer multiple of q. So the string $0^{4(p-q)+\ell q - p}10^q$ has no tuple representation.

Our goal is to demonstrate that, in any reduction starting from $10^{k(p-q)+q}$, the first extraordinary step takes place only as late as the completion of the loop.

We are now going to construct two functions h, h' that estimate the length of the shortest ordinary reduction to the next string that has the factor $10^{(p-q)\ell}$, and the length of the shortest extraordinary reduction, respectively. These two functions will be based on the following auxiliary functions g_k.

DEFINITION 6.6. *The functions $g_k : \mathbb{N}^{k+1} \to \mathbb{N}$, $k \in \mathbb{N}$ are defined by*

$$g_k(x_0, \dots, x_k) = \frac{1}{r-1}(r^{x_1+\dots+x_k} + r^{x_2+\dots+x_k} + \dots + r^{x_k} - k) \ .$$

g_k does not depend on its first argument. This is intentional.

The following derived properties will be useful below.

PROPOSITION 6.7. *For all $k \geq 1, x_0, \dots, x_k$ the following hold:*

1. $g_k(x_0, \dots, x_k) = g_{k-1}(x_0, \dots, x_{k-1})$ *if* $x_k = 0$;
2. $g_k(x_0, \dots, x_k) \geq g_{k-1}(x_1, \dots, x_k)$;
3. $g_k(x_0, \dots, x_k) = g_{k+r-1}(x_0, \dots, x_i + 1, \underbrace{0, \dots, 0}_{r-1}, x_{i+1} - 1, \dots, x_k) + 1$ *for all* $0 \leq i \leq k-1$ *such that* $x_{i+1} \geq 1$;
4. $g_k(x_0, \dots, x_k) \geq g_k(x_0 + 1, x_1, \dots, x_{i-1}, x_i - 1, x_{i+1}, \dots, x_k)$ *for all* $1 \leq i \leq k$ *such that* $x_i \geq 1$;
5. g_k *is monotone in each argument.*

The next lemma is the workhorse of this section. It states that a reduction step decreases by at most one, the minimal value of all those g_k terms which have the same sum of arguments.

LEMMA 6.8. *Let $s \to s'$ be an ordinary step where $T(s) = (x_0, \dots, x_k; y_0, \dots, y_k)$ and $T(s') = (x'_0, \dots, x'_{k'}; y'_0, \dots, y'_{k'})$. Then for every $1 \leq i' \leq j' \leq k'$, $z'_{j'} \leq x'_{j'}$ there exist $1 \leq i \leq j \leq k$, $z_j \leq x_j$ such that*

$$\begin{gathered} g_{j'-i'}(x'_{i'}, \dots, x'_{j'-1}, z'_{j'}) \geq g_{j-i}(x_i, \dots, x_{j-1}, z_j) - 1, \\ x'_{i'} + \dots + x'_{j'-1} + z'_{j'} = x_i + \dots + x_{j-1} + z_j \ . \end{gathered}$$

Proof. Let $0 \leq m \leq k-1$ be determined by the rewrite step $s \to_m s'$. Then by Proposition 6.5 we have $k' = k + r - 1$, $x'_0 = x_0$, $\dots$, $x'_{m-1} = x_{m-1}$, $x'_m = x_m + 1$, $x'_{m+1} = \dots = x'_{m+r-1} = 0$, $x'_{m+r} = x_{m+1} - 1$, $x'_{m+r+1} = x_{m+2}$, $\dots$, $x'_{k'} = x_k$. And we have $y'_0 = y_0$, $\dots$, $y'_{m-1} = y_{m-1}$, $y'_m = y_m + 1$, $y'_{m+1} = \dots = y'_{m+r-1} = 0$, $y'_{m+r} = y_{m+1}$, $y'_{m+r+1} = y_{m+2}$, $\dots$, $y'_{k'} = y_k$.

Let $1 \leq i' \leq j' \leq k'$ and $z'_{j'} \leq x'_{j'}$. The proof is done by case analysis on i' and j'.

Case 1: $1 \leq i' \leq j' \leq m$. If $j' \neq m$ or $z'_{j'} \neq x_m + 1$ then choose $i = i', j = j', z_j = z'_{j'}$. In this case we get

$$g_{j'-i'}(x'_{i'}, \dots, x'_{j'-1}, z'_{j'}) = g_{j-i}(x_i, \dots, x_{j-1}, z_j) \ . \tag{6.1}$$

Else choose $i = i', j = m+1, z_j = 1$. Here we use $x_{m+1} > 0$ to establish $z_j \le x_j$. We get

$$\begin{aligned}
g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) &= g_{m-i}(x_i, \ldots, x_{m-1}, x_m + 1) \\
&= g_{m+r-i}(x_i, \ldots, x_{m-1}, x_m + 1, \underbrace{0, \ldots, 0}_{r}) \\
&= g_{m+1-i}(x_i, \ldots, x_m, 1) - 1 \\
&= g_{j-i}(x_i, \ldots, x_{j-1}, z_j) - 1,
\end{aligned}$$

by Items 1 and 3 of Proposition 6.7.

Case 2: $m+1 \le i' \le j' \le m+r-1$. Then $x_{i'} + \cdots + x_{j'-1} + z_{j'} = 0$. Choose any $1 \le i \le k$, and let $j = i$ and $z_j = 0$. Then

$$g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) = 0 = g_0(0) \ .$$

Case 3: $m + r \le i' \le j' \le k'$. If $i' \ne m+r$ or $j' = i'$ then choose $i = i' - r + 1, j = j' - r + 1, z_j = z'_{j'}$. We get (6.1). Else we have $i' = m + r$ and $j' > i'$, and so $x'_{i'} = x_{m+1} - 1$. In this case let $i = m+1$. If $z'_{j'} > 0$ then let j and z_j be defined by $j = j' - r + 1$ and $z_j = z'_{j'} - 1$; else let $i \le j < j' - r + 1$ be the greatest number such that $x_j > 0$ and let $z_j = x_j - 1$. By $x_i > 0$, j and z_j are well-defined. Thus we get

$$\begin{aligned}
g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) &= g_{j'-r+1-i}(x_i - 1, x_{i+1}, \ldots, x_{j'-r}, z'_{j'}) \\
&= g_{j-i}(x_i - 1, x_{i+1}, \ldots, x_{j-1}, z_j + 1) \\
&\ge g_{j-i}(x_i, x_{i+1}, \ldots, x_{j-1}, z_j)
\end{aligned}$$

by Items 1 and 4 of Proposition 6.7.

Case 4: $1 \le i' \le m$ and $m + r \le j' \le k'$. Choose $i = i', j = j' - r + 1, z_j = z'_{j'}$. We get

$$\begin{aligned}
g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) &= g_{j-i+r-1}(x_i, \ldots, x_m + 1, \underbrace{0, \ldots, 0}_{r-1}, x_{m+1} - 1, \ldots, x_{j-1}, z_j) \\
&= g_{j-i}(x_i, \ldots, x_m, x_{m+1}, \ldots, x_{j-1}, z_j) - 1
\end{aligned}$$

by Item 3 of Proposition 6.7. Note that if $j' = m + r$ then $x'_{j'} = x_{m+1} - 1 = x_j - 1$. So one can always choose $z_j = z'_{j'}$ yielding $z_j = z'_{j'} \le x'_{j'} \le x_j$ as required.

Case 5: $1 \le i' \le m < j' \le m + r - 1$. This case reduces to Case 1 by the identity

$$g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) = g_{m-i'}(x'_{i'}, \ldots, x'_m)$$

due to Item 1 of Proposition 6.7.

Case 6: $m + 1 \le i' < m + r \le j' \le k'$. This case reduces to Case 3 by the inequality

$$g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) \ge g_{j'-m-r}(x'_{m+r}, \ldots, x'_{j'-1}, z'_{j'})$$

due to Item 2 of Proposition 6.7.

These are all cases. In each case it is easy to show that $x'_{i'} + \cdots + x'_{k'} = x_i + \cdots + x_k$. This finishes the proof. □

Definition 6.9. *Let $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$ and let $x_1 + \cdots + x_k \ge \ell$. Then $h(s) \in \mathbb{N}$ is defined by*

$$h(s) = \min\{g_{j-i}(x_i, \ldots, x_{j-1}, z_j) \mid 1 \le i \le j \le k, z_j \le x_j, x_i + \cdots + x_{j-1} + z_j = \ell\} \ .$$

Well-definedness of $h(s)$ follows immediately from the fact that the minimum is taken from a finite, non-empty set.

LEMMA 6.10. *Let $s \to s'$ be an ordinary step where $T(s) = (x_0, \dots, x_k; y_0, \dots, y_k)$ and $T(s') = (x'_0, \dots, x'_{k'}; y'_0, \dots, y'_{k'})$. If $x'_1 + \dots + x'_{k'} \geq \ell$ then $x_1 + \dots + x_k \geq \ell$ and $h(s) \leq h(s') + 1$.*

Proof. The condition $x'_1 + \dots + x'_{k'} \geq \ell$ ensures that $h(s')$ is defined. If $s \to_m s'$ for $1 \leq m \leq k-1$ then $x_1 + \dots + x_k = x'_1 + \dots + x'_{k'} \geq \ell$ by Proposition 6.5. Else $s \to_m s'$ for $m = 0$ and then $(x_1 - 1) + \dots + x_k = x'_1 + \dots + x'_{k'} \geq \ell$. So $x_1 + \dots + x_k \geq \ell$ whence $h(s)$ is defined.

By definition of $h(s')$, there is $1 \leq i' \leq j' \leq k'$, $z'_{j'} \leq x'_{j'}$ such that both $h(s') = g_{j'-i'}(x'_{i'}, \dots, x'_{j'-1}, z'_{j'})$ and $x'_{i'} + \dots + x'_{j'-1} + z'_{j'} = \ell$. Hence by Lemma 6.8, there is $1 \leq i \leq j \leq k$, $z_j \leq x_j$ such that

$$h(s') \geq g_{j-i}(x_i, \dots, x_{j-1}, z_j) - 1 \quad \text{and} \quad x_i + \dots + x_{j-1} + z_j = \ell \ .$$

So $h(s) \leq g_{j-i}(x_i, \dots, x_{j-1}, z_j) \leq h(s') + 1$. □

LEMMA 6.11. *Let $T(s) = (x_0, \dots, x_k; y_0, \dots, y_k)$. If $s \to^n u10^{(p-q)\ell}v$ for some strings u, v is an ordinary reduction then $x_1 + \dots + x_k \geq \ell$ and $h(s) \leq n$.*

Proof. By induction on n. The base case $n = 0$ is proven by $h(s) \leq g_0(\ell) = 0$. For the inductive step let $s \to s' \to^{n-1} u10^{(p-q)\ell}v$, let $x'_1 + \dots + x'_{k'} \geq \ell$, and let $h(s') \leq n-1$. Thus $x_1 + \dots + x_k \geq \ell$ and $h(s) \leq n$ by Lemma 6.10. □

With Lemma 6.11 we have a criterion for ordinary reductions. For extraordinary reductions we pursue a similar line of reasoning. We start with a lemma akin to Lemma 6.8. If $i = j$ then for convenience let $g_{j-i}(y_i, x_{i+1}, \dots, x_{j-1}, z_j) = g_0(z_j)$ and let $y_i + x_{i+1} + \dots + x_{j-1} + z_j = z_j$. Note that we require $z_j \leq y_j$ if $i = j$ and $z_j \leq x_j$ else.

LEMMA 6.12. *Let $s \to s'$ be an ordinary step where $T(s) = (x_0, \dots, x_k; y_0, \dots, y_k)$ and $T(s') = (x'_0, \dots, x'_{k'}; y'_0, \dots, y'_{k'})$. Then for every $1 \leq i' < j' \leq k'$, $z'_{j'} \leq x'_{j'}$ and for every $1 \leq i' = j' \leq k'$, $z'_{j'} \leq y'_{j'}$ there exist $1 \leq i < j \leq k$, $z_j \leq x_j$ or $1 \leq i = j \leq k$, $z_j \leq y_j$ such that*

$$\begin{aligned} g_{j'-i'}(y'_{i'}, x'_{i'+1}, \dots, x'_{j'-1}, z'_{j'}) &\geq g_{j-i}(y_i, x_{i+1}, \dots, x_{j-1}, z_j) - 1, \\ y'_{i'} + x'_{i'+1} + \dots + x'_{j'-1} + z'_{j'} &= y_i + x_{i+1} + \dots + x_{j-1} + z_j \ . \end{aligned}$$

Proof. Let $0 \leq m \leq k-1$ be determined by the rewrite step $s \to_m s'$. Case 1: $1 \leq i' = j' \leq k'$, $z'_{j'} \leq y'_{j'}$. If $j' \neq m$ or $z'_{j'} \neq z_m + 1$ then $g_0(z'_{j'}) = 0 = g_0(z_j)$. Else choose $j = m+1, z_j = 1$. Here we use $y_{m+1} > 0$ to establish $z_j < y_j$. We get

$$g_0(z'_{j'}) = g_r(y_m + 1, \underbrace{0, \dots, 0}_{r}) = g_1(y_m, 1) - 1 = g_{j-i}(y_i, z_j) - 1$$

by Items 1 and 3 of Proposition 6.7.

Case 2: $1 \leq i' < j' \leq k'$, $z'_{j'} \leq x'_{j'}$.

Case 2.1: $m + 1 \leq i' < j' \leq m + r - 1$. Then $y_{i'} + x_{i'+1} + \dots + x_{j'-1} + z_{j'} = 0$. Choose any $1 \leq i \leq k$, and let $j = i$ and $z_j = 0$. Then

$$g_{j'-i'}(y'_{i'}, x'_{i'+1}, \dots, x'_{j'-1}, z'_{j'}) = 0 = g_0(0) \ .$$

Case 2.2: $i' = m + r$. Choose $= i' - r + 1, j = j' - r + 1, z_j = z'_{j'}$. We get

$$\begin{aligned} g_{j'-i'}(y'_{i'}, x'_{i'}, \dots, x'_{j'-1}, z'_{j'}) &= g_{j-i}(y_i, x_{i+1}, \dots, x_{j-1}, z_j), \\ y'_{i'} + x'_{i'} + \dots + x'_{j'-1} + z'_{j'} &= y_i + x_{i+1} + \dots + x_{j-1} + z_j \ . \end{aligned}$$

Case 2.3: $1 \le i' \le m$ or $i' \neq m+r$ and $m+r-1 \le j' \le k'$. We carry over the proof of Lemma 6.8, observing the facts $i' \neq j'$, $i \neq j$, and $y'_{i'} - x'_{i'} = y_i - x_i$. Then we may conclude from the proof of Lemma 6.8 that

$$\begin{aligned} g_{j'-i'}(y'_{i'}, x'_{i'+1}, \ldots, x'_{j'-1}, z'_{j'}) &= g_{j'-i'}(x'_{i'}, \ldots, x'_{j'-1}, z'_{j'}) \\ &\geq g_{j-i}(x_i, \ldots, x_{j-1}, z_j) - 1 \\ &= g_{j-i}(y_i, x_{i+1}, \ldots, x_{j-1}, z_j) - 1 \end{aligned}$$

and

$$\begin{aligned} y'_{i'} + x'_{i'+1} + \cdots + x'_{j'-1} + z'_{j'} &= (y'_{i'} - x'_{i'}) + x'_{i'} + \cdots + x'_{j'-1} + z'_{j'} \\ &= (y'_{i'} - x'_{i'}) + x_i + \cdots + x_{j-1} + z_j \\ &= (y_i - x_i) + x_i + \cdots + x_{j-1} + z_j \\ &= y_i + x_{i+1} + \cdots + x_{j-1} + z_j \ . \end{aligned}$$

This finishes the proof. □

LEMMA 6.13. *Let $s \to s'$ be an ordinary step where $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$ and $T(s') = (x'_0, \ldots, x'_{k'}; y'_0, \ldots, y'_{k'})$. If $y'_i + x'_{i+1} + \cdots + x'_{k'} \geq \ell$ for some $1 \le i' \le k'$ then $y_i + x_{i+1} + \cdots + x_k \geq \ell$ for some $1 \le i \le k$.*

Proof. The claim immediately follows from the following claim. For every $1 \le i' \le k'$ there is $1 \le i \le k$ such that

$$\Delta = y_i + x_{i+1} + \cdots + x_k - (y'_{i'} + x'_{i'+1} + \cdots + x'_{k'}) \geq 0 \ .$$

The proof is done by case analysis on i'.

Case 1: $1 \le i' \le m-1$. Choose $i = i'$. Then

$$\Delta = x_m + x_{m+1} - (x'_m + \underbrace{0 + \cdots + 0}_{r-1} + x'_{m+r}) = x_m + x_{m+1} - (x_m + 1 + x_{m+1} - 1) = 0 \ .$$

Case 2: $i' = m$. Again choose $i = i'$. Then

$$\Delta = y_m + x_{m+1} - (y'_m + \underbrace{0 + \cdots + 0}_{r-1} + x'_{m+r}) = y_m + x_{m+1} - (y_m + 1 + x_{m+1} - 1) = 0 \ .$$

Case 3: $m+1 \le i' \le m+r-1$. Choose $i = m+1$. Then $\Delta = x_{m+1} - x'_{m+r} = 1 \geq 0$.

Case 4: $m+r \le i' \le k'$. Choose $i = i' - r + 1$. Then obviously $\Delta = 0$. This finishes the proof. □

DEFINITION 6.14. *Let $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$ and let $y_i + x_{i+1} + \cdots + x_k \geq \ell$ for some $1 \le i \le k$. Then $h'(s) \in \mathbb{N}$ is defined by*

$$h'(s) = \min\{g_{j-i}(y_i, x_{i+1}, \ldots, x_{j-1}, z_j) \mid 1 \le i < j \le k, z_j \le x_j, y_i + x_{i+1} + \cdots + x_{j-1} + z_j = \ell\} \ .$$

Because the minimum is taken from a finite, non-empty set, $h'(s)$ is well-defined.

Using Lemma 6.12 and 6.13, in the same way as Lemma 6.10, one can prove:

LEMMA 6.15. *Let $s \to s'$ be an ordinary step where $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$ and $T(s') = (x'_0, \ldots, x'_{k'}; y'_0, \ldots, y'_{k'})$. If $y'_i + x'_{i+1} + \cdots + x'_{k'} \geq \ell$ for some $1 \le i' \le k'$ then $y_i + x_{i+1} + \cdots + x_k \geq \ell$ for some $1 \le i \le k$, and $h'(s) \le h'(s') + 1$.*

Thus we get a lemma like Lemma 6.11:

LEMMA 6.16. *Let $T(s) = (x_0, \ldots, x_k; y_0, \ldots, y_k)$. If there is an extraordinary reduction $s \to^n t$ then $y_i + x_{i+1} + \cdots + x_k \geq \ell$ for some $1 \leq i \leq k$, and moreover $h'(s) \leq n$.*

Proof. Let $s \to^n s'$ be shortest, i.e. $s \to^{n-1} s'$ is an ordinary reduction and only the last step $s' \to t$ is extraordinary. The inductive base $n = 1$ is proven by $h'(s) \leq g_1(\ell - 1, 1) = 1$. For the inductive step $s \to s'' \to^{n-2} s'$ let $y'_i + x'_{i+1} + \cdots + x'_{k'} \geq \ell$ for some $1 \leq i' \leq k'$ and let $h'(s'') \leq n - 1$. Thus $y_i + x_{i+1} + \cdots + x_k \geq \ell$ for some $1 \leq i \leq k$ and $h'(s) \leq n$ by Lemma 6.15. □

Now let us prove that the length of the loop in Lemma 4.6 is minimal.

LEMMA 6.17. *R admits no loops of length less than $1 + \sum_{i=0}^{\ell-1} r^i$ for $\ell = \lceil \frac{p}{q} \rceil$.*

Proof. Suppose that $s \to^+ usv$ is a loop of minimal length. By Theorem 5.2 we may assume that $s \to^+ usv$ is a forward closure. By Lemma 5.3, $s = 10^{(p-q)k+q}$ for some $k \geq 0$. By Lemma 5.6, $k \geq 1 + \ell$. Since the loop is a forward closure, all zeroes in s must be consumed during the reduction. By $OVL(\ell, \ell) = \emptyset$ we may assume that the steps are rearranged to $s \to^k (0^p 1^{r-1})^k 10^q = s' \to^n usv$. Case 1: $s' \to^n usv$ is ordinary. Then we get $h(s') \leq n$ by Lemma 6.11 and by $10^{(p-q)\ell}$ prefix of s. We compute $h(s')$ as follows:

$$\begin{aligned}
h(s') &= g_{(r-1)(\ell-1)}(1, \underbrace{\underbrace{0, \ldots, 0}_{r-2}, 1, \underbrace{0, \ldots, 0}_{r-2}, 1, \ldots, \underbrace{0, \ldots, 0}_{r-2}}_{\ell-1}, 1) \\
&= \frac{1}{r-1}((r-1)r^{\ell-1} + (r-1)r^{\ell-2} + \cdots + (r-1)r^1 - (r-1)(\ell-1)) \\
&= \sum_{i=0}^{\ell-1} r^i - \ell\ .
\end{aligned}$$

So the total length of the reduction is $k + n \geq k + h(s') = k + \sum_{i=0}^{\ell-1} r^i - \ell \geq 1 + \sum_{i=0}^{\ell-1} r^i$. Case 2: $s' \to^n usv$ is extraordinary. Then we get $h'(s') \leq n$ by Lemma 6.16. It turns out that $h'(s') = h(s)$. So, no matter whether the reduction $s \to^n usv$ is ordinary or not, we get $k + n \geq 1 + \sum_{i=0}^{\ell-1} r^i$. This finishes the proof. □

Acknowledgements. I am grateful to Robert McNaughton, David Musser, and Paliath Narendran for their encouragement to pursue this research and to Hanne Gottliebsen for reading the manuscript.

REFERENCES

[1] R. BOOK AND F. OTTO, *String-rewriting systems*, Texts and Monographs in Computer Science, Springer, New York, 1993.

[2] N. DERSHOWITZ, *Termination of linear rewriting systems*, in Proc. 8th Int. Coll. Automata, Languages and Programming, LNCS 115, Springer, 1981, pp. 448–458.

[3] ——, *Termination of rewriting*, J. Symb. Comput., 3 (1987), pp. 69–115. Corrigendum: 4, 3, Dec. 1987, 409-410.

[4] A. GESER, *Note on normalizing, non-terminating one-rule string rewriting systems*, Theoret. Comput. Sci., 243 (2000), pp. 489–498.

[5] ——, *Decidability of termination of "grid" string rewriting rules*, SIAM J. Comput., (2001). Accepted for publication.

[6] A. GESER AND H. ZANTEMA, *Non-looping string rewriting*, Theoret. Informatics Appl., 33 (1999), pp. 279–301.

[7] M. HERMANN, *Divergence des systèmes de réécriture et schématisation des ensembles infinis de termes*, habilitation, Université de Nancy, France, Mar. 1994.

[8] G. HUET AND D. S. LANKFORD, *On the uniform halting problem for term rewriting systems*, Tech. Report 283, INRIA, Rocquencourt, FR, Mar. 1978.

[9] M. JANTZEN, *Confluent string rewriting*, vol. 14 of EATCS Monographs on Theoretical Computer Science, Springer, Berlin, 1988.

[10] Y. KOBAYASHI, M. KATSURA, AND K. SHIKISHIMA-TSUJI, *Termination and derivational complexity of confluent one-rule string rewriting systems*, Theoret. Comput. Sci., 262 (2001), pp. 583–632.

[11] W. KURTH, *Termination und Konfluenz von Semi-Thue-Systemen mit nur einer Regel*, dissertation, Technische Universität Clausthal, Germany, 1990.

[12] D. S. LANKFORD AND D. R. MUSSER, *A finite termination criterion*, tech. report, Information Sciences Institute, Univ. of Southern California, Marina-del-Rey, CA, 1978.

[13] Y. MATIYASEVITCH AND G. SÉNIZERGUES, *Decision problems for semi-Thue systems with a few rules*, in IEEE Symp. Logic in Computer Science'96, 1996.

[14] R. MCNAUGHTON, *The uniform halting problem for one-rule Semi-Thue Systems*, Tech. Report 94-18, Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, Aug. 1994. See also "Correction to 'The Uniform Halting Problem for One-rule Semi-Thue Systems'," unpublished paper, Aug. 1996.

[15] ——, *Well-behaved derivations in one-rule Semi-Thue Systems*, Tech. Report 95-15, Dept. of Computer Science, Rensselaer Polytechnic Institute, Troy, NY, Nov. 1995. See also "Correction by the author to 'Well-behaved derivations in one-rule Semi-Thue Systems'," unpublished paper, July 1996.

[16] ——, *Semi-Thue Systems with an Inhibitor*, J. Automated Reasoning, 26 (1997), pp. 409–431.

[17] P. NARENDRAN AND F. OTTO, *The problems of cyclic equality and conjugacy for finite complete rewriting systems*, Theoret. Comput. Sci., 47 (1986), pp. 27–38.

[18] M. J. O'DONNELL, *Computing in systems described by equations*, LNCS 58, Springer, 1977.

[19] G. SÉNIZERGUES, *On the termination problem for one-rule Semi-Thue Systems*, in RTA-96, LNCS 1103, Springer, 1996, pp. 302–316.

[20] K. SHIKISHIMA-TSUJI, M. KATSURA, AND Y. KOBAYASHI, *On termination of confluent one-rule string rewriting systems*, Inform. Process. Lett., 61 (1997), pp. 91–96.

[21] H. ZANTEMA AND A. GESER, *A complete characterization of termination of* $0^p1^q \to 1^r0^s$, Applicable Algebra in Engineering, Communication, and Computing, 11 (2000), pp. 1–25.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	February 2002	Contractor Report

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Loops of superexponential lengths in one-rule string rewriting	C NAS1-97046 WU 505-90-52-01
6. AUTHOR(S) Alfons Geser	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) ICASE Mail Stop 132C NASA Langley Research Center Hampton, VA 23681-2199	8. PERFORMING ORGANIZATION REPORT NUMBER ICASE Report No. 2002-3
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-2199	10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA/CR-2002-211451 ICASE Report No. 2002-3

11. SUPPLEMENTARY NOTES

Langley Technical Monitor: Dennis M. Bushnell
Final Report
To be submitted to the International Conference on Rewriting Techniques and Applications (RTA).

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Unclassified–Unlimited Subject Category 60, 61 Distribution: Nonstandard Availability: NASA-CASI (301) 621-0390	

13. ABSTRACT *(Maximum 200 words)*

Loops are the most frequent cause of non-termination in string rewriting. In the general case, non-terminating, non-looping string rewriting systems exist, and the uniform termination problem is undecidable. For rewriting with only one string rewriting rule, it is unknown whether non-terminating, non-looping systems exist and whether uniform termination is decidable. If in the one-rule case, non-termination is equivalent to the existence of loops, as McNaughton conjectures, then a decision procedure for the existence of loops also solves the uniform termination problem. As the existence of loops of bounded lengths is decidable, the question is raised how long shortest loops may be. We show that string rewriting rules exist whose shortest loops have superexponential lengths in the size of the rule.

14. SUBJECT TERMS	15. NUMBER OF PAGES
string rewriting, semi-Thue system, uniform termination, termination, loop, one-rule, single-rule	16
	16. PRICE CODE A03

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified		